REMARKS

In accordance with the foregoing, the specification has been amended to improve form. Claims 1-14 are pending and under consideration. No new matter is presented in this Amendment.

REJECTIONS UNDER 35 U.S.C. §101:

Claims 9 is rejected under 35 U.S.C. §101 because claim 9 is directed to nonstatutory subject matter. Specifically, claim 9 is directed to a recording medium in a form of a signal. Accordingly, the specification has been amended to remove "carrier wave" as an example of a computer readable recording medium. Therefore, the Applicants respectfully request that the rejection be withdrawn.

REJECTIONS UNDER 35 U.S.C. §102:

Claims 1-14 are rejected under 35 U.S.C. §102(b) as being anticipated by Dang (U.S. Patent 7,216,300). The Applicants respectfully traverse the rejection and request reconsideration.

Regarding the rejection of independent claim 1, it is noted that claim 1 recites "determining whether the applet is a bound applet or an unbound applet" and "if the applet is an unbound applet, immediately issuing predetermined commands to the virtual machine to first set the unbound applet into an initiate state and then into a start state." In contrast, Dang teaches a method of executing an applet, whereby the applet is initiated without determining whether the applet is a bound applet or an unbound applet (FIG. 4, operations 404 through 408). That is, in the entirety of Dang, there is no suggestion of determining whether the applet is a bound applet or an unbound applet, and the method taught by Dang is performed in the same manner for both bound and unbound applets. Specifically, after the applet is initiated, if another initiate function is called, the method merely ignores the function call (FIG. 4, operation 406). Similarly, after the applet is started, if another start function is called, the method merely reopens the already-started applet window (FIG. 4, operation 412 and 416). Furthermore, if a stop function or a destroy function is called, the method closes the applet window or deallocates resources, respectively, regardless of whether the applet is a bound applet or an unbound applet (FIG. 4, operations 422 through 430). Thus, the method taught by Dang is applied uniformly to both bound and unbound applets, with no determination made. The Examiner cites column 2, lines

49-52 as a disclosure of determining whether the applet is bound or unbound. However, column 2, lines 49-52 of Dang discloses an applet that continues to execute when the browser's focus switches to another web page without any suggestion of determining whether the applet is bound or unbound. That is, the entirety of Dang, including column 2, lines 49-52 cited by the Examiner, does not suggest any operation of determining whether the applet is bound or unbound. Therefore, the Applicants respectfully submit that Dang fails to disclose, implicitly or explicitly, a determination of whether an applet is bound or unbound, as recited in claim 1.

Regarding the rejection of claim 2, it is noted that this claim depends from claim 1 and is, therefore, allowable for at least the reasons set forth above.

Regarding the rejection of claim 3, it is noted that this claim depends from claim 1 and is, therefore, allowable for at least the reasons set forth above. Furthermore, it is noted that claim 3 recites that if it is "determined by the browser that a markup document connected to the bound applet has a grammatically correct structure, issuing a predetermined command to the virtual machine to set the bound applet into an initiate state." In contrast, Dang does not recite any determination of whether the markup document has a grammatically correct structure, as recited in claim 3.

Regarding the rejection of claim 4, it is noted that this claim depends from claim 3 and is, therefore, allowable for at least the reasons set forth above.

Regarding the rejection of independent claim 5, it is noted that claim 5 recites an application manager that "determines whether the loaded applet is a bound applet or an unbound applet" and "if the loaded applet is an unbound applet, immediately issues predetermined commands to the virtual machine to first set the loaded unbound applet into an initiate state and then into a start state." In contrast, Dang teaches a method of executing an applet, whereby the applet is initiated without determining whether the applet is a bound applet or an unbound applet (FIG. 4, operations 404 through 408). That is, in the entirety of Dang, there is no suggestion of determining whether the applet is a bound applet or an unbound applet, and the method taught by Dang is performed in the same manner for both bound and unbound applets. Specifically, after the applet is initiated, if another initiate function is called, the method merely ignores the function call (FIG. 4, operation 406). Similarly, after the applet is started, if another start function is called, the method merely reopens the already-started applet window (FIG. 4, operation 412 and 416). Furthermore, if a stop function or a destroy function is called, the method closes the applet window or deallocates resources, respectively, regardless

of whether the applet is a bound applet or an unbound applet (FIG. 4, operations 422 through 430). Thus, the method taught by Dang is applied uniformly to both bound and unbound applets, with no determination made. Therefore, the Applicants respectfully submit that Dang fails to disclose, implicitly or explicitly, a determination of whether an applet is bound or unbound, as recited in claim 5.

Regarding the rejection of claim 6, it is noted that this claim depends from claim 5 and is, therefore, allowable for at least the reasons set forth above.

Regarding the rejection of claim 7, it is noted that this claim depends from claim 5 and is, therefore, allowable for at least the reasons set forth above. Furthermore, it is noted that claim 7 recites "the browser informs the application manager that the markup document connected to the bound applet has a grammatically correct structure." In contrast, Dang does not recite any informing of whether the markup document has a grammatically correct structure, as recited in claim 7.

Regarding the rejection of claim 8, it is noted that this claim depends from claim 7 and is, therefore, allowable for at least the reasons set forth above.

Regarding the rejection of independent claim 9, it is noted that claim 9 recites "determining whether the requested applet is a bound applet or an unbound applet" and "if the requested applet is an unbound applet, immediately issuing predetermined commands to the virtual machine to first set the requested loaded unbound applet into an initiate state and then into a start state." In contrast, Dang teaches a method of executing an applet, whereby the applet is initiated without determining whether the applet is a bound applet or an unbound applet (FIG. 4, operations 404 through 408). That is, in the entirety of Dang, there is no suggestion of determining whether the applet is a bound applet or an unbound applet, and the method taught by Dang is performed in the same manner for both bound and unbound applets. Specifically, after the applet is initiated, if another initiate function is called, the method merely ignores the function call (FIG. 4, operation 406). Similarly, after the applet is started, if another start function is called, the method merely reopens the already-started applet window (FIG. 4, operation 412 and 416). Furthermore, if a stop function or a destroy function is called, the method closes the applet window or deallocates resources, respectively, regardless of whether the applet is a bound applet or an unbound applet (FIG. 4, operations 422 through 430). Thus, the method taught by Dang is applied uniformly to both bound and unbound applets, with no determination made. The Examiner cites column 2, lines 49-52 as a disclosure of determining whether the

applet is bound or unbound. However, column 2, lines 49-52 of Dang discloses an applet that continues to execute when the browser's focus switches to another web page without any suggestion of determining whether the applet is bound or unbound. That is, the entirety of Dang, including column 2, lines 49-52 cited by the Examiner, does not suggest any operation of determining whether the applet is bound or unbound. Therefore, the Applicants respectfully submit that Dang fails to disclose, implicitly or explicitly, a determination of whether an applet is bound or unbound, as recited in claim 9.

Regarding the rejection of independent claim 10, it is noted that claim 10 recites a processing of "a markup document classifying tagged applets into bound and unbound applets," and "determining whether an applet execution of the markup document is a bound applet or an unbound applet according to the classifying." In contrast, Dang teaches a method of executing an applet, whereby the applet is initiated without classification or determining whether the applet is a bound applet or an unbound applet (FIG. 4, operations 404 through 408). That is, in the entirety of Dang, there is no suggestion of a markup document classifying tagged applets or of determining whether the applet is a bound applet or an unbound applet, and the method taught by Dang is performed in the same manner for both bound and unbound applets. Specifically, after the applet is initiated, if another initiate function is called, the method merely ignores the function call (FIG. 4, operation 406). Similarly, after the applet is started, if another start function is called, the method merely reopens the already-started applet window (FIG. 4, operation 412 and 416). Furthermore, if a stop function or a destroy function is called, the method closes the applet window or deallocates resources, respectively, regardless of whether the applet is a bound applet or an unbound applet (FIG. 4, operations 422 through 430). Thus, the method taught by Dang is applied uniformly to both bound and unbound applets, with no classification or determination made. Therefore, the Applicants respectfully submit that Dang fails to disclose, implicitly or explicitly, a classification of an applet or a determination of whether an applet is bound or unbound, as recited in claim 10.

Regarding the rejection of claims 11-12, it is noted that these claims depend from claim 10 and are, therefore, allowable for at least the reasons set forth above.

Regarding the rejection of independent claim 13, it is noted that claim 13 recites "classifying tagged applets of a markup document," and "controlling different execution life cycles of the tagged applets according to the classifying." In contrast, Dang teaches a method of executing an applet, whereby the applet is initiated without classification (FIG. 4, operations 404 through 408). That is, in the entirety of Dang, there is no suggestion of classifying tagged

applets of a markup document, and the method taught by Dang is performed in the same manner for all types of applets. Specifically, after the applet is initiated, if another initiate function is called, the method merely ignores the function call (FIG. 4, operation 406). Similarly, after the applet is started, if another start function is called, the method merely reopens the already-started applet window (FIG. 4, operation 412 and 416). Furthermore, if a stop function or a destroy function is called, the method closes the applet window or deallocates resources, respectively, regardless of whether the applet is a bound applet or an unbound applet (FIG. 4, operations 422 through 430). Thus, the method taught by Dang is applied uniformly all types of applets, with no classification made. Therefore, the Applicants respectfully submit that Dang fails to disclose, implicitly or explicitly, a classification of an applet, as recited in claim 13.

Regarding the rejection of claim 14, it is noted that this claim depends from claim 13 and is, therefore, allowable for at least the reasons set forth above.

CONCLUSION:

It has been confirmed with the Examiner through telephone conversation that the Office Action mailed January 24, 2008 is a non-Final Office Action. There being no further outstanding objections or rejections, it is submitted that the application is in condition for allowance. An early action to that effect is courteously solicited.

Finally, if there are any formal matters remaining after this response, the Examiner is requested to telephone the undersigned to attend to these matters.

If there are any additional fees associated with filing of this Amendment, please charge the same to our Deposit Account No. 503333.

Respectfully submitted,

STEIN, MCEWEN & BUI, LLP

Date: 4/22/08

By: _____
Michael D. Stein
Registration No. 37,240

1400 Eye St., N.W.
Suite 300
Washington, D.C. 20005
Telephone: (202) 216-9505
Facsimile: (202) 216-9510